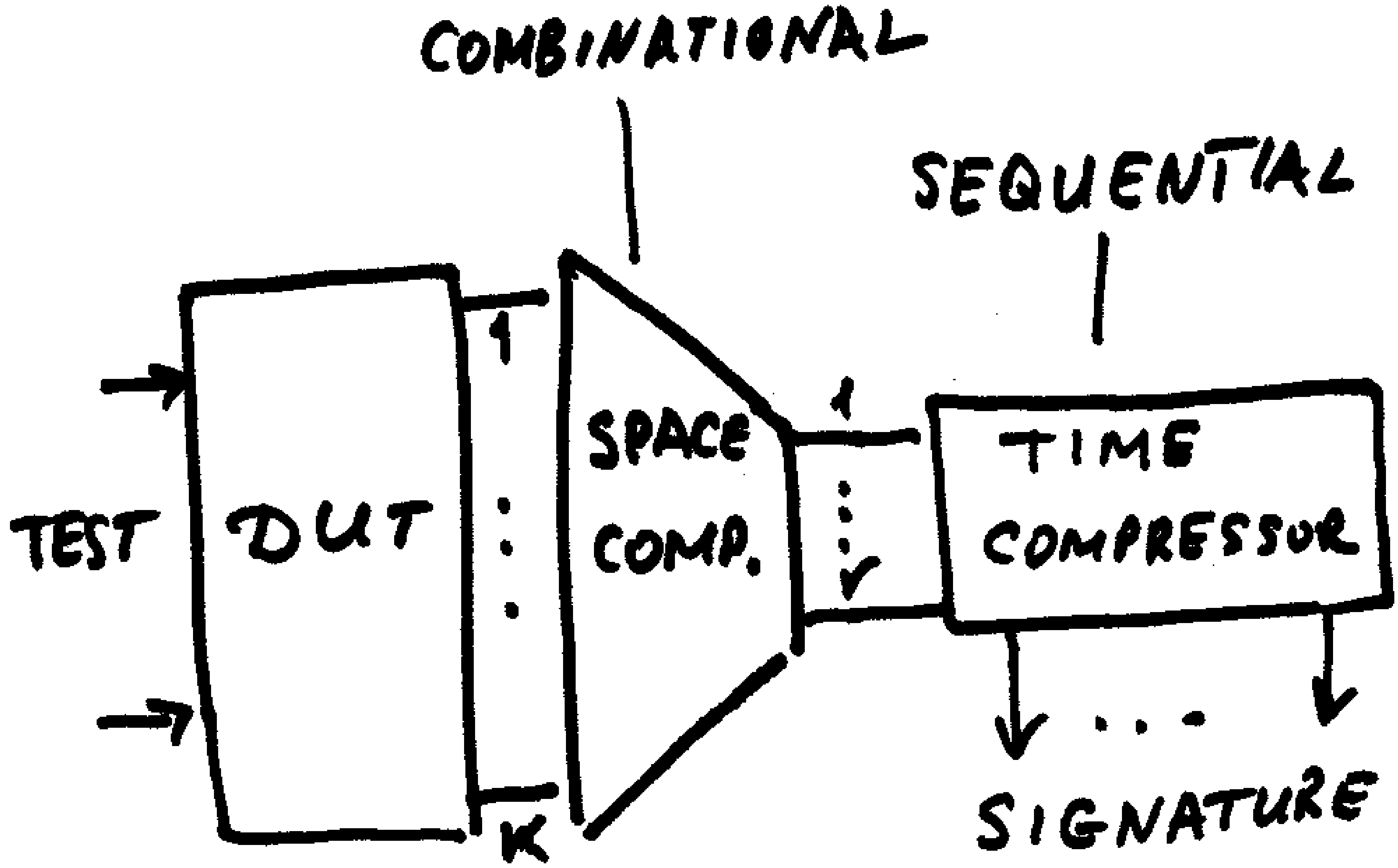
SPACE-TIME COMPRESSION



TIME COMPRESSORS ARE LFSR, TRANSITION COUNT, ADDER/ ACCUMULATOR.

PROBLEMS:

1. min $r(K, \ell)$

2. CONSTRUCT SC.

3. OVERHEAD

4. PROBABILITY OF MASKING

LINEAR COMPRESSORS $\Rightarrow$ XOR gates only $\Leftrightarrow$ DECODERS OF LINEAR ERROR-DETECTING CODES

Problems: 1) Estimate min r for a given K and ℓ (multiplicity of errors at the output of DUT)

$r(K, \ell) = ?$

2) Design optimal SC with

$r = r(K, \ell)$

Solutions in:

K. K. Saluja, M. G. Karpovsky, "Testing Computer Hardware Through Data Compression in Space and Time" Proc. Int. Test Conf. 1983

M. G. Karpovsky, S. R. Reddy, K. K. Saluja, "A data compression Technique for Built-in Self Test", Proc. 1985 Fault Tolerant Computing Symposium, 1985.

Estimations on a minimal number of observation points

Lower bound

$$r(k,\ell) \geq \left\lceil \log_2 \left(\sum_{i=0}^{\lfloor \ell/2 \rfloor} \binom{k}{i} + \varepsilon \right) \right\rceil$$

$$\varepsilon = \begin{cases} 0, & \ell\text{-even} \\ \binom{k-1}{\lceil \ell/2 \rceil}, & \ell \text{ odd} \end{cases}$$

Example 2 $k=6$, $\ell=3$ $\Rightarrow \lceil \ell/2 \rceil = 2$,

$$\varepsilon = \binom{5}{2} = 10$$

$$r(6,3) \geq \lceil \log_2 (1+6+10) \rceil = 5$$

$k=32$, $\ell=3$

$$r(32,3) \geq \left\lceil \log_2 \left(1+32+\binom{31}{2}\right) \right\rceil = 9$$

Upper Bound

$$r(k,\ell) \leq \left\lceil \log_2 \sum_{i=0}^{\ell} \binom{k-1}{i} \right\rceil$$

$$\ell > 2$$

$$r(k,1) = 1$$
$$r(k,2) = \lceil \log_2 (k+1) \rceil$$

Example 3 $\quad k = 9,\ \ell = 3$

$$r(9,3) \leq 7$$

$r = 32$, $\approx \ell = 3$

$$9 \leq K(32,3) \leq \left\lceil \log_2 \left(1 + 31 + \binom{31}{2} + \binom{31}{3} \right\rceil = 13$$

Design of Space Compressors by Linear Error Correcting Codes

Let H be a check matrix of a code $V = (K, K-r)$ with distance $l+1$. (For any $v \in V$ $Hv = 0$)

Let $Re(H)$ be a network computing syndrome for this code

$Re(H)$ has K inputs and r outputs uses XOR gates only

Then $\boxed{SC = Re(H)}$

Output of a space compressor may be applied to the input of a time compressor (ex. LFSF with parallel load)

Error: $\underline{Y} \mapsto \underline{Y}^*$ $\quad e = \underline{Y} \oplus \underline{Y}^*$ $\quad l = \|e\|$ - multiplicity of error

$S = H\underline{Y}^* = H(\underline{Y} \oplus e) = H\underline{Y} \oplus He$ - error detected iff $He \neq 0 \Rightarrow$

$\Rightarrow H$ is a check matrix of a code detecting l-errors

If there exists $(K, K-r)$ perfect code with distance $\ell+1$, then

$$r(K,\ell) = \log_2 \sum_{i=0}^{\lfloor \ell/2 \rfloor} \binom{K}{i}$$

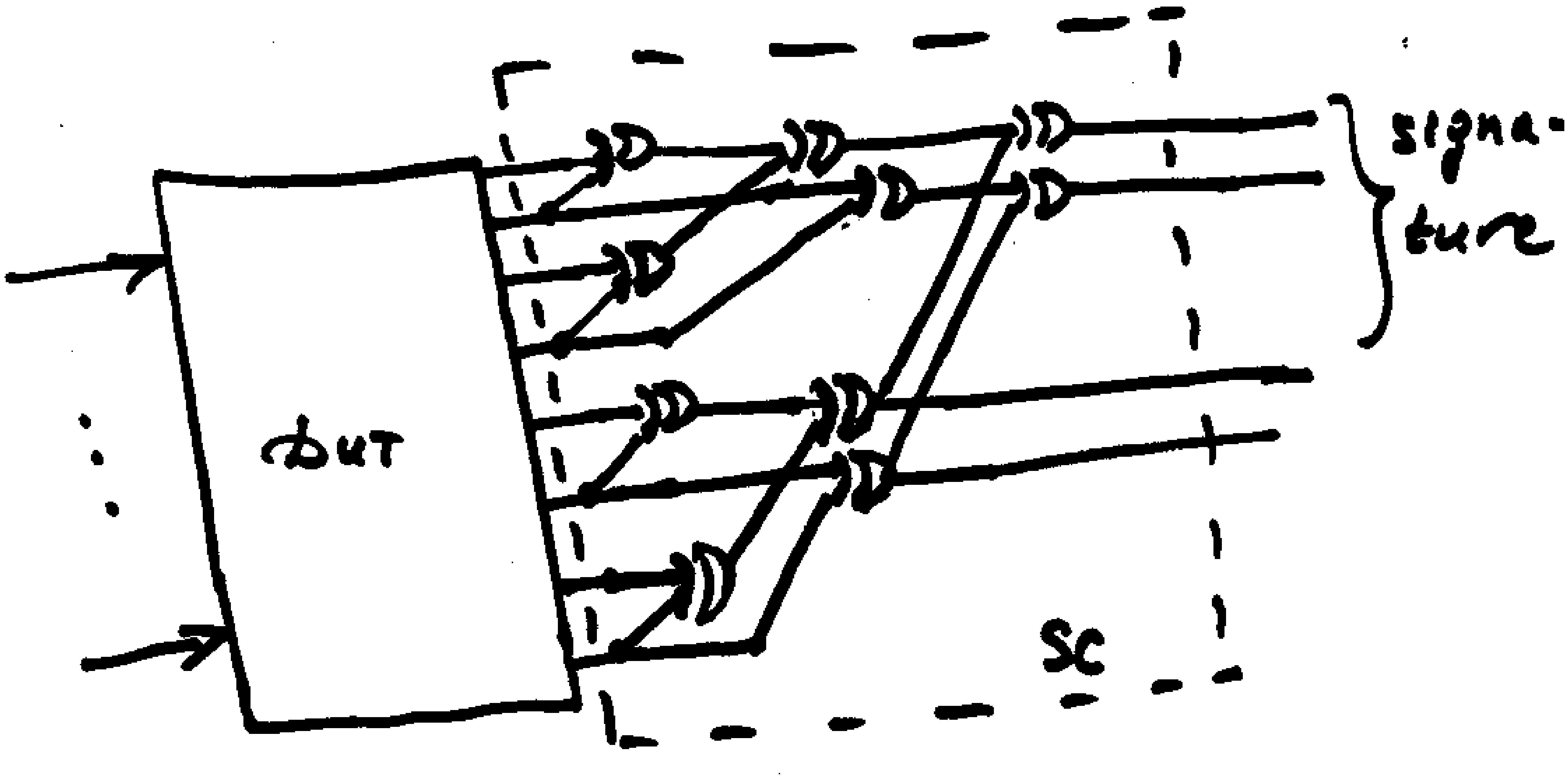
Ex 4.
Since there exists (23, 12) Golay code which is perfect with distance 7 we have

$r(23, 6) = 11$ - number of check bits in the code.

<u>Example 5</u> $k = 8$, $\ell = 3$

Let us use (8,4) extended Hamming code with $dist = \ell + 1 = 4$

$$H = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$ then



the SC based on (8,4) extended Hamming Code

joint minimization of functions computing signature. Faults in SC are detected

Bounds on Minimal Numbers of Observation Points $r(K,\ell)$ and the Corresponding Codes

ℓ	K	$r(K,\ell)$ Lower Bound	$r(K,\ell)$ Upper Bound	Code
1	K	1	1	Parity checking
2	2^i-1	i	i	Hamming code
3	K	$1+\lceil \log_2 K \rceil$	$1+\lceil \log_2 (K+1) \rceil$	Extended Hamming code
4	2^i-1	$2i-1$	$2i$	BCH code
6	23	11	11	Golay code

for errors with multiplicity $> \ell$ the probability of masking an error in SC is

$$P = \frac{2^{k-r}-1}{2^k-1} \simeq 2^{-r}$$

Errors uniformly distributed

Example 6 MC 68020
$K=116$ control outputs for ROM

take $\ell=10$ and $(116,84)$ code with dist 11

Then we have 32-bit signature $r=32$

All errors with multiplicity ≤ 10 detected; for errors with multiplicity ≥ 11 probability of masking $\simeq 2^{-32}$

SC may be: 1) part of a tester
2) SC on a chip.

Gate-Counts for Space compressors

$L(K,\ell)$ – number of 2-input XOR gates

Ex. $L(K,1) = K-1$

$$L(K,\ell) \leq \lceil \tfrac{1}{2}(K-r)r + K-1 \rceil$$

$$r = \lceil \log_2 \sum_{i=0}^{\ell} \binom{K-1}{i} \rceil$$

For large K

$$L(K,\ell) \simeq \frac{Kr}{\log_2 K}$$

Space Compressors Detecting (Propogating) a Given Set of Errors

Let C is a code of length $n=K+r$ detecting set E of errors and H is a check matrix for C

then $\text{Re}(H) = SC$

and $SC(z \oplus e) \neq SC(z)$

for any $e \in E$.

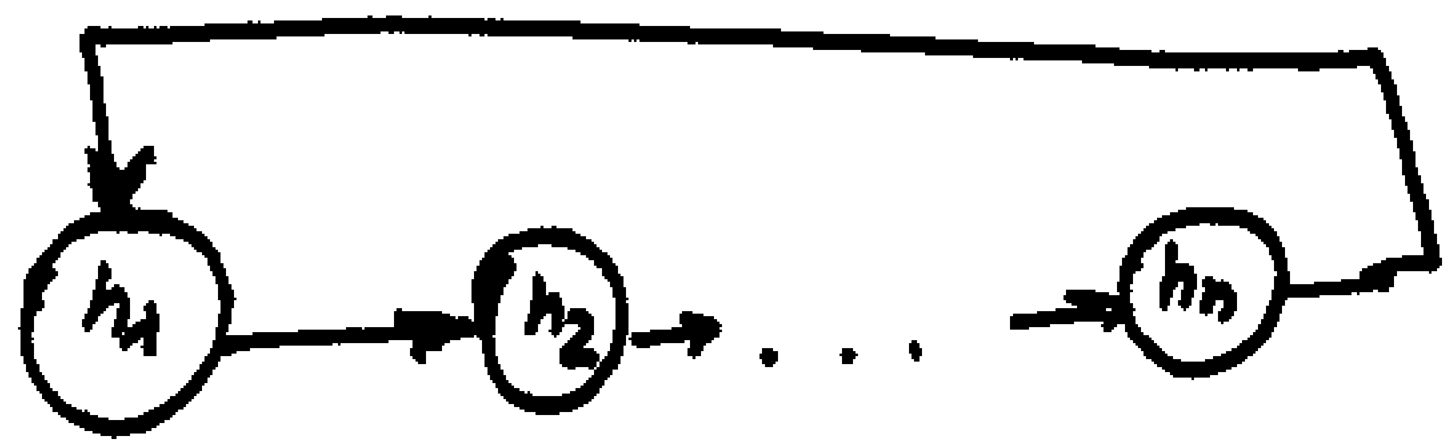
Time Compressors

Let $H = [h_1 h_2 \dots h_n]$ $\quad h_i \in \{0,1\}^r$

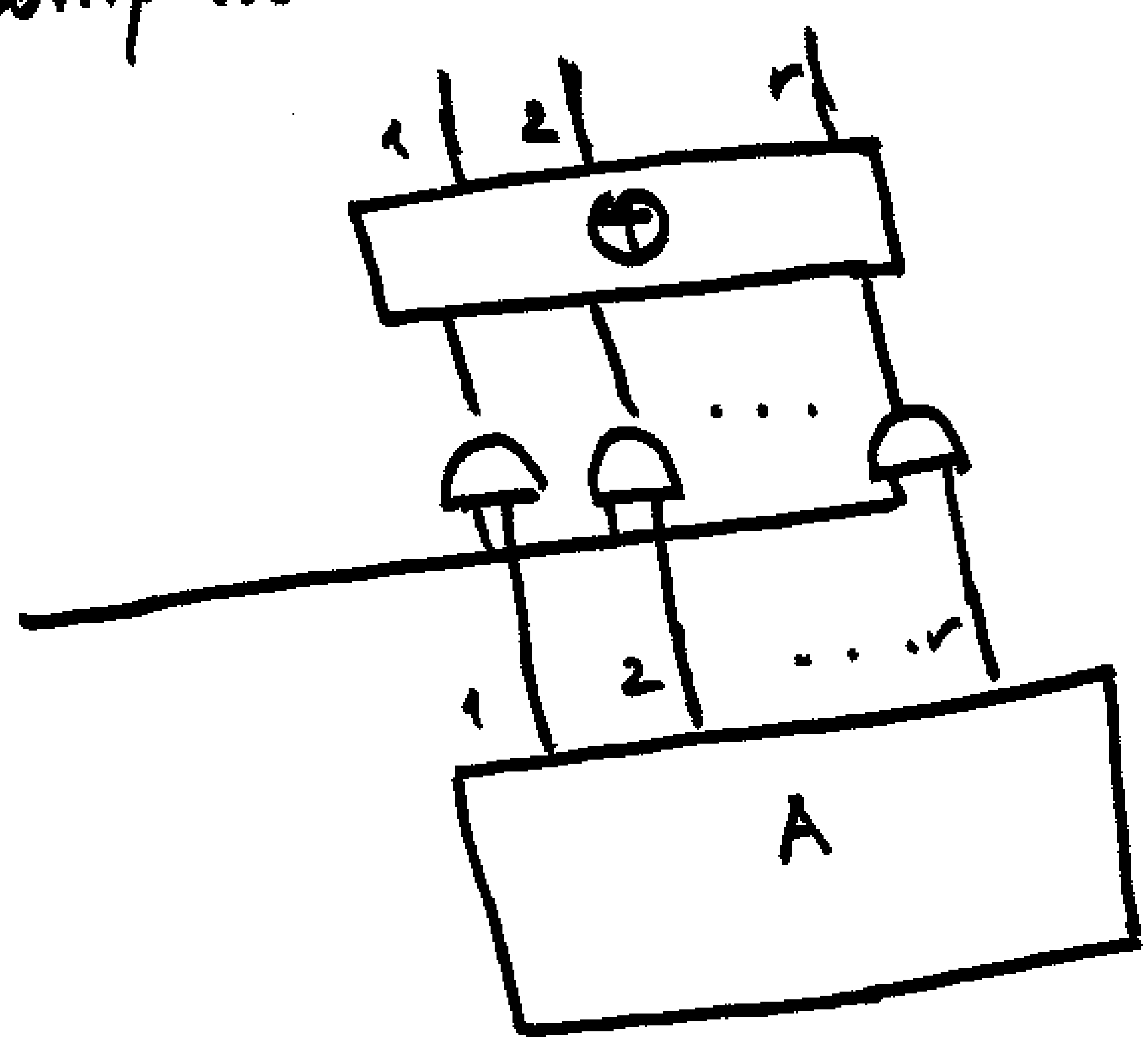
then $S = Hz = h_1 \cdot z_1 \oplus h_2 z_2 \oplus \dots \oplus h_n z_n$ $\quad z_i \in \{0,1\}$.

$= \bigoplus\limits_i h_i z_i$

let A is a <u>autonomous</u> sequential Network with r FFs with transition diagram



then for have for the time compressor



Space Compressors for Specific Classes of Faults

Errors in instruction decoding:

'no decoding' or 'multiple decoding' $\Rightarrow \ell = 1$

'wrong decoding' $\Rightarrow \ell = 2$

SC is a syndrome calculator for the Hamming code with $dist = 3$

If $n = 2^m$ then $r = m+1$

only $m+1$ observation points

Errors in ALU

$l=1$ single arithmetical errors

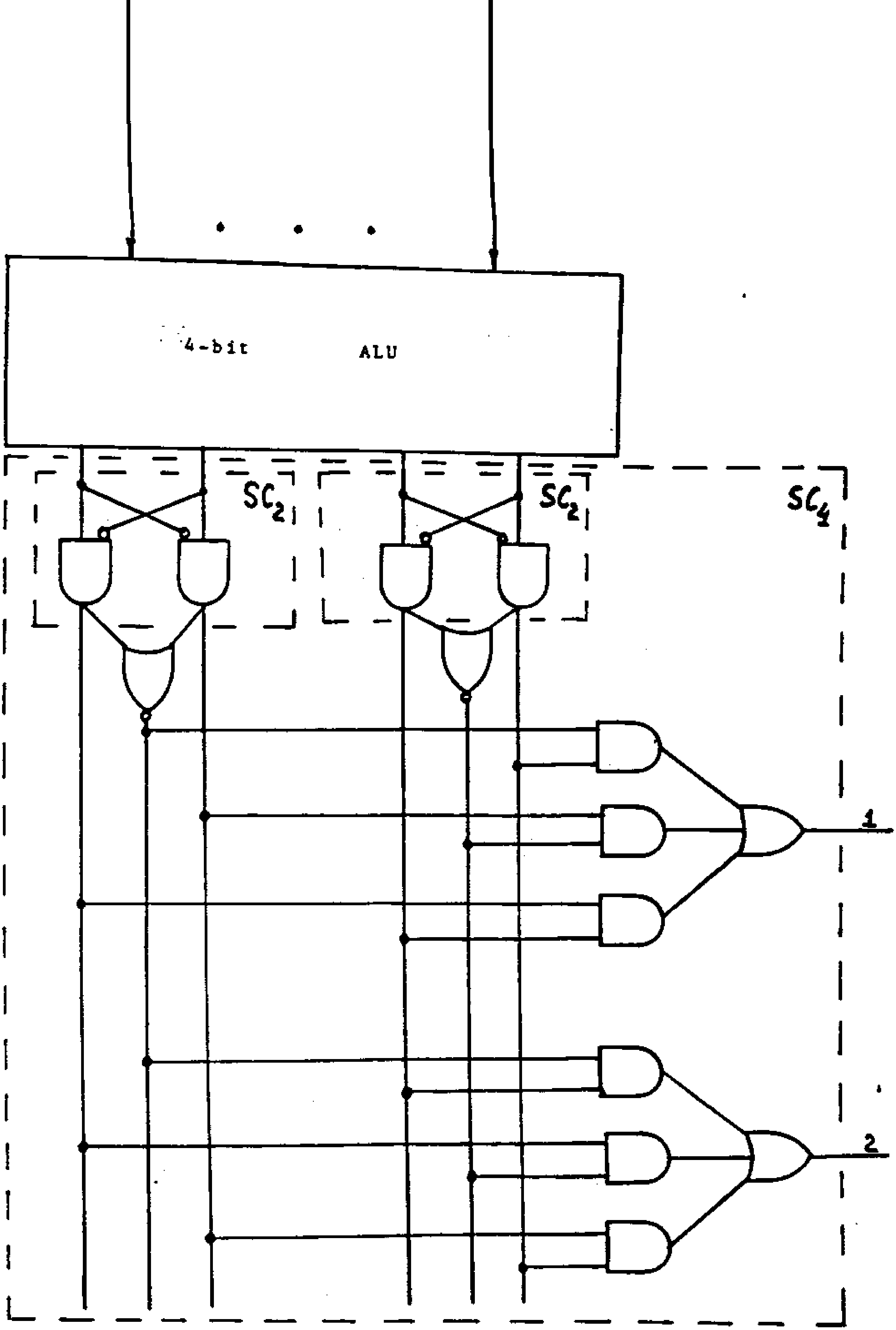
SC is a decoder of the 3N arithmetical code computing a residue mod 3. for an output of ALU

$\boxed{r=2}$

Gate counts (equivalent 2-input gates) $L(n)$ for detection of single errors in n-bit ALU

n	4	8	16	32	64
$L(n)$	16	44	100	212	436

$$r = \lceil \log_2 A \rceil .$$



139

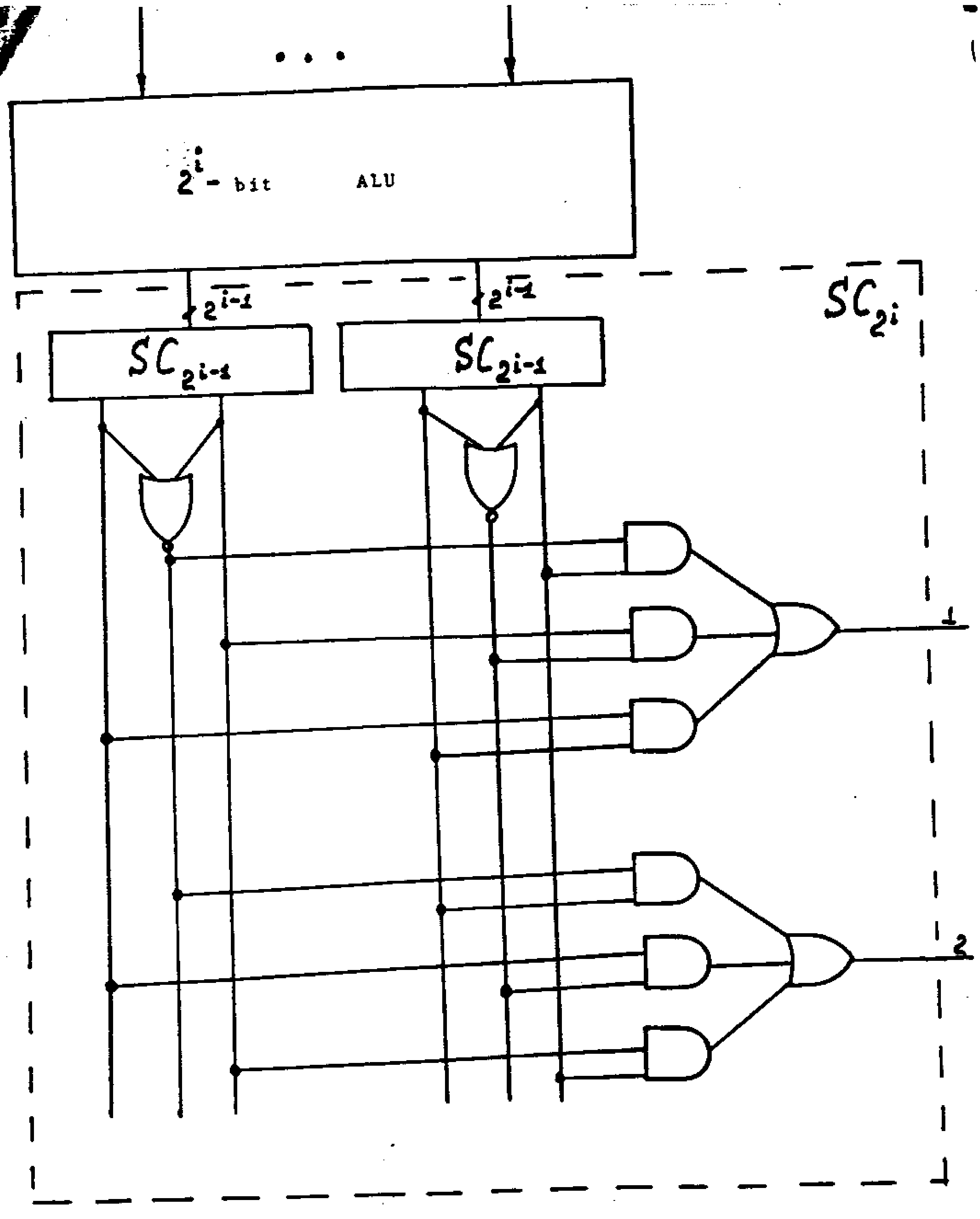
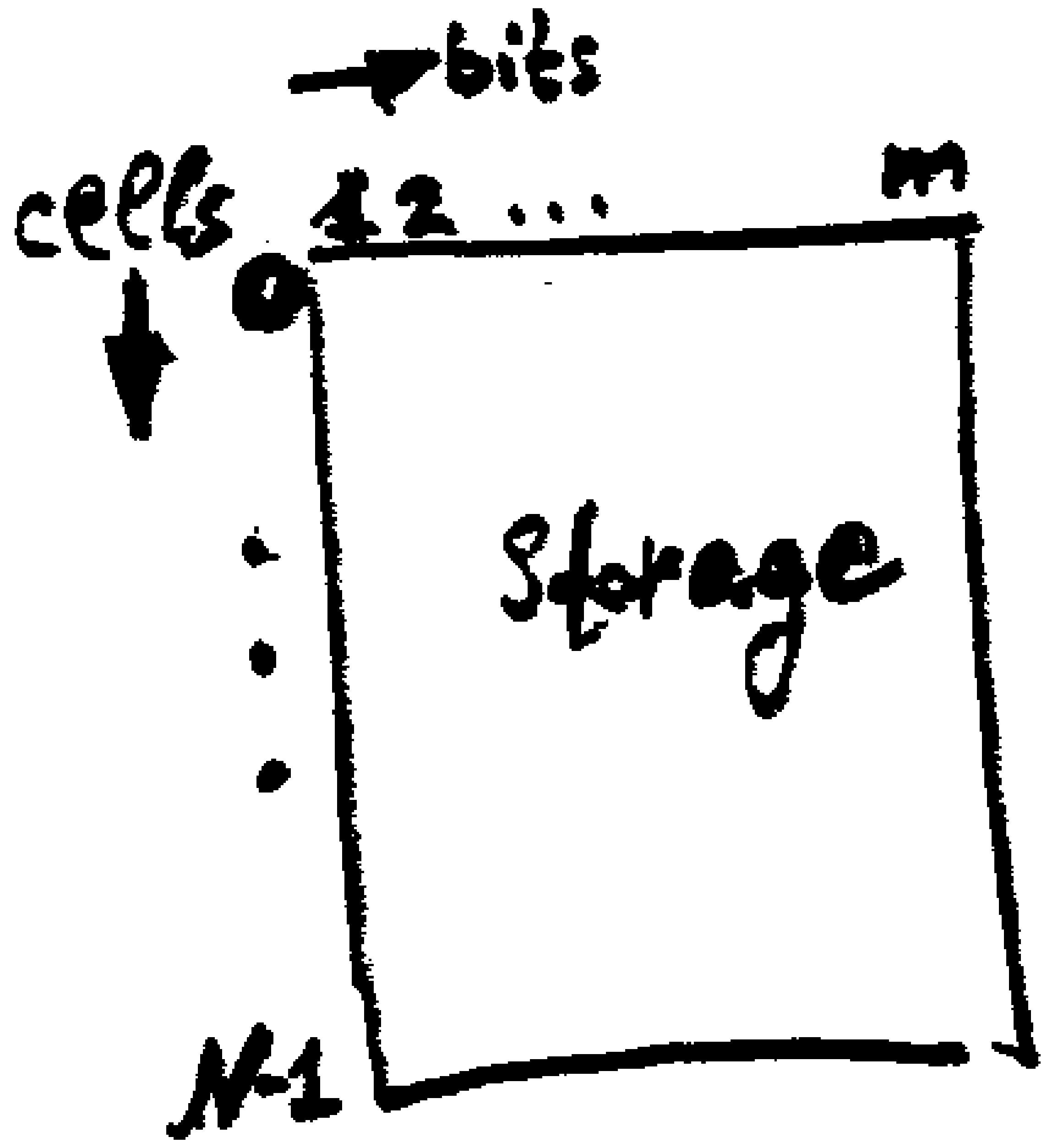


Fig. 6 Recursive procedure for the implementation of the SC with 2^i inputs by two SCs with 2^{i-1} inputs.

SC753 VOL8 141 8

Check Sums for Memory Testing (ROM)



For every column:

$$\text{Checksum} = \sum_{i=0}^{N-1} W_i b_i$$

W_i - weights

b_i - binary value of ith bit

If $W_i = 2^i \Rightarrow$ duplication

Take $W_i = i+1$.

Checksums are efficient for ROMs and not very efficient for RAMs (Not suitable for on-line testing)

Example 1.

$m=4$, $N=4$.

ROM

$$\begin{array}{c}\begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}\end{array} \begin{array}{l} W_i \\ \to 1 \\ \to 2 \\ \to 3 \\ \to 4 \end{array}$$

Checksum for the first column: $1+3+4=8$

For the second column: $2+3+4=9$

For the third column: $1+2+3=6$

For the fourth column: 1

Example 2

$8K \times 8$ ROM $\Rightarrow$

26 bits for each checksum

Overhead: $26 \times 8 = 208$ bits

Space Overhead

$$\text{Checksum}_{max} = \sum_{i=1}^{N} i = \frac{N(N+1)}{2}$$

$$\text{Space overhead} = m\left\lceil \log_2 \frac{N(N+1)}{2} \right\rceil = \simeq 2m\log_2 N$$

For the previous example $N = 8K$, $m = 8$

$$\text{space overhead} = 8 \cdot \left\lceil \log_2 \frac{8K \cdot 8K}{2} \right\rceil = 208$$

DETECTION OF SINGLE AND DOUBLE ERRORS

CORRECTION OF SINGLE ERRORS

IN ONE COLUMNS